



SOLID - Single Responsibility

Programação Orientada a Objeto II

SOLID - Single Responsibility

Programação Orientada a Objeto II

"Nunca deve haver mais de uma razão para alterar uma classe. Em outras palavras, toda classe deve ter somente uma responsabilidade."

Essa é uma definição carregada popularmente ao longo do tempo, mas que está errada. A definição correta é:

**Um módulo deve ser responsável por um, e apenas um, ator.
(Robert C. Martin - Arquitetura Limpa)**

É tentador associar considerar como ator aquele sobre quem realizamos uma ação. Mas aqui, deveríamos falar sobre aquele quem demandou a mudança e que realiza a ação. Vamos detalhar mais a fundo no exemplo a seguir.

SOLID - Single Responsibility

Programação Orientada a Objeto II

Exemplo:

Considere uma aplicação de escola e a representação de aluno como:

```
public class Aluno {  
    private String nome;  
    private String email;  
    //getters, setters, constructors  
}
```


SOLID - Single Responsibility

Programação Orientada a Objeto II

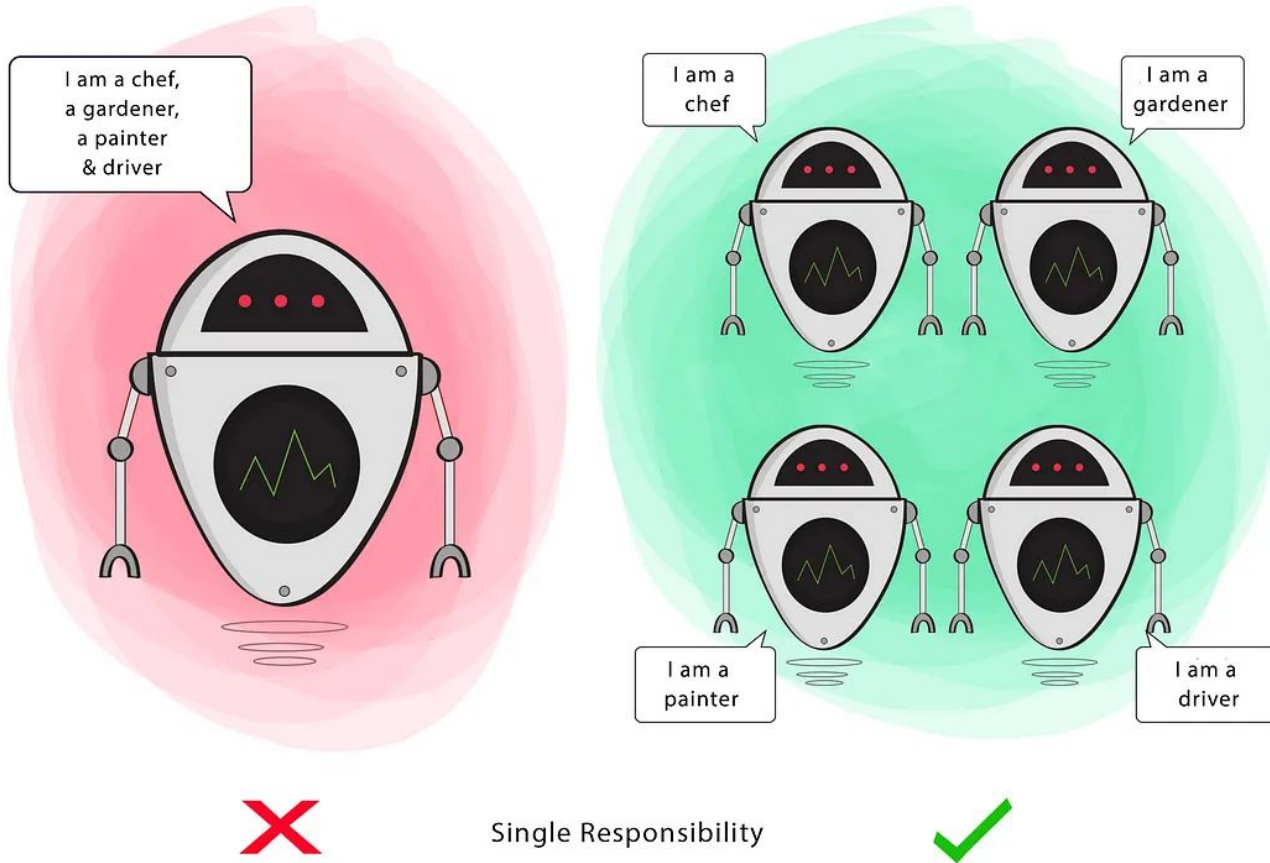
Exemplo:

Agora, vamos adicionar algumas funcionalidades relacionadas ao aluno.

```
public class Aluno {  
    public void enviarEmail (String mensagem) {  
        //code  
    }  
    public void pagarMensalidade (BigDecimal valor) {  
        //code  
    }  
    public double realizarAvaliacao (String mensagem) {  
        //code  
    }  
}
```

SOLID - Single Responsibility

Programação Orientada a Objeto II



Obrigada